Dizajni i Softuerit

Tema 2 Procesi dhe Menaxhimi i Dizajnimit të Softuerit

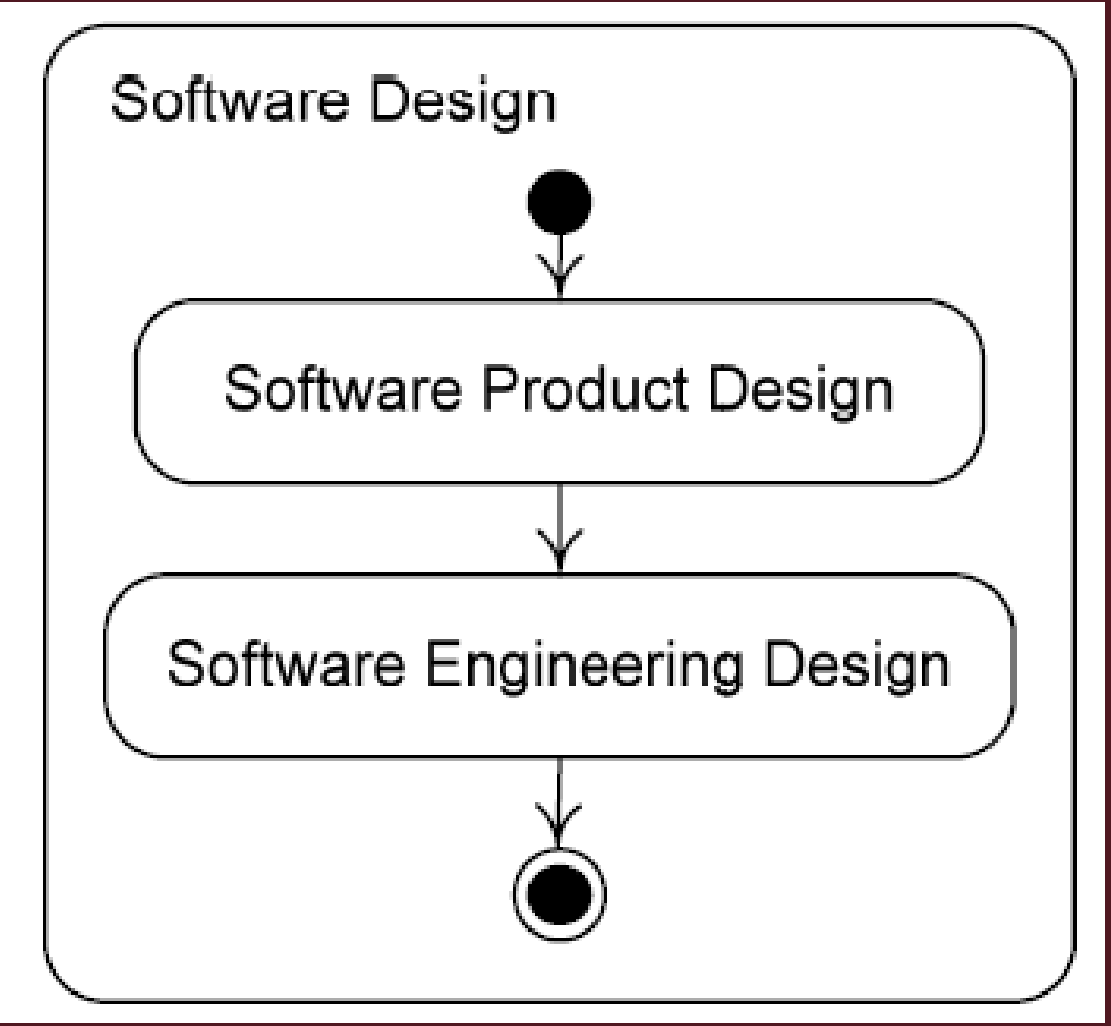
Edmond Jajaga, PhD

Lecture 1 Summary

- Software design is important.
- Software design is best thought of as problem solving.
- Abstraction is a fundamental design technique.
- Modeling (which relies on abstraction) is a basic design tool.
- Software design comprises both product and engineering design.
  - Product design occurs mainly in the requirements specification phase; engineering design mainly in the design and implementation phases.
- OO analysis and design methods are now dominant.

Lecture 2:
Software Design Processes and Management

Software Design Process (Recall)



Let’s refine this very abstract description of the software design process...

Objectives

- To understand how design consists of **analysis** and **resolution** activities
- To illustrate and explain **generic processes** for software product and engineering design
- To explain the five main tasks of **project management**
- To understand **iterative** planning and tracking
- To see how to **apply project management** principles to software design projects

Topics

- Analysis and resolution
- Generic problem-solving and design processes
- Generic software product and engineering design processes
- Project management
- Iterative planning and tracking
- Applying project management to software design projects

Analysis and Resolution

- **Confusion** arises around the term *design*.
- This confusion is removed by adopting the following terminology.

Analysis is breaking down a design problem to understand it.

Resolution is solving a design problem.

Analysis and Resolution

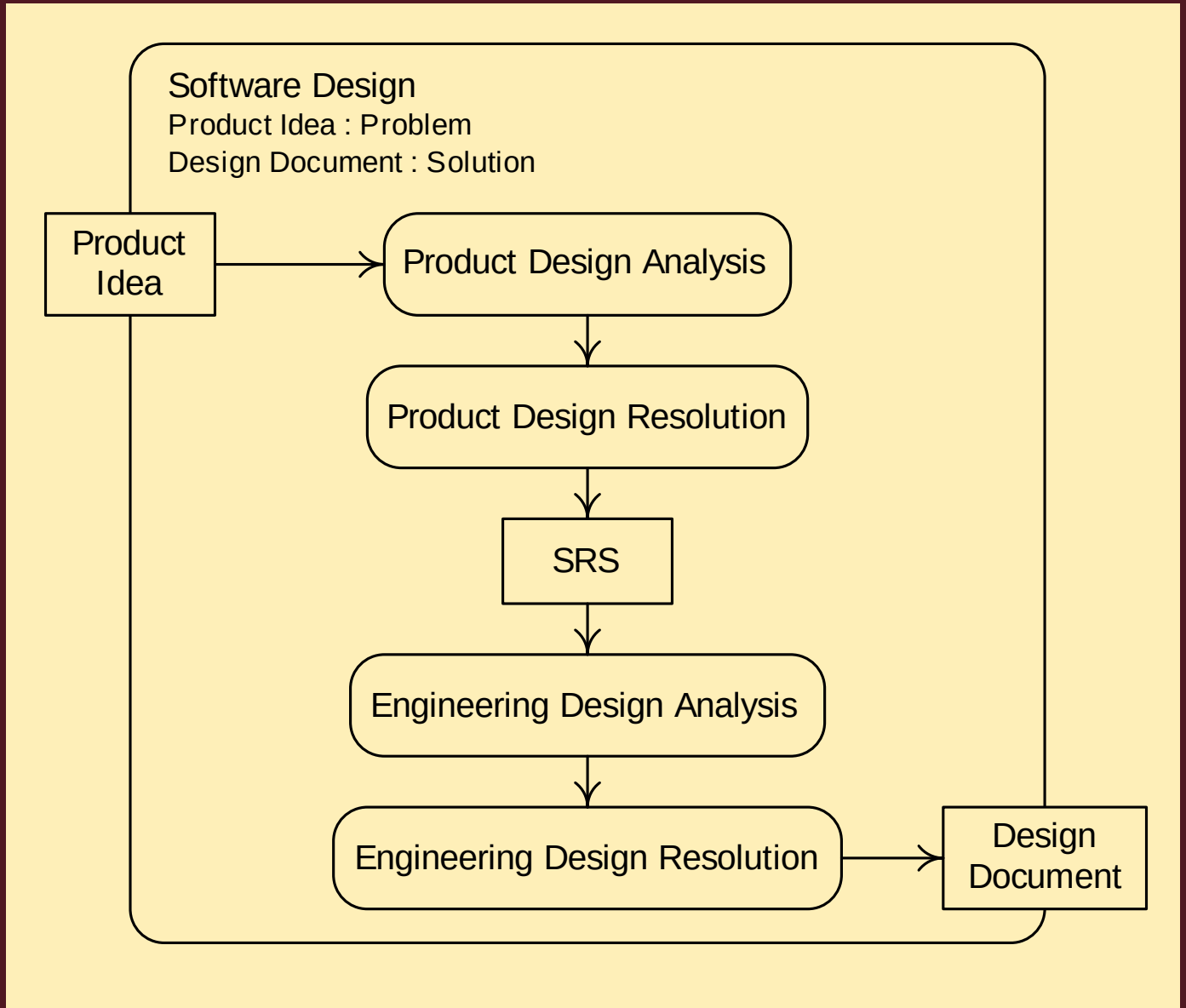
- **Confusion** arises around the term *design*.
- This confusion is removed by adopting the following terminology.

Analysis is breaking down a design problem to understand it.

Resolution is solving a design problem.

Design = Analysis + Resolution

Analysis and Resolution in Software Design

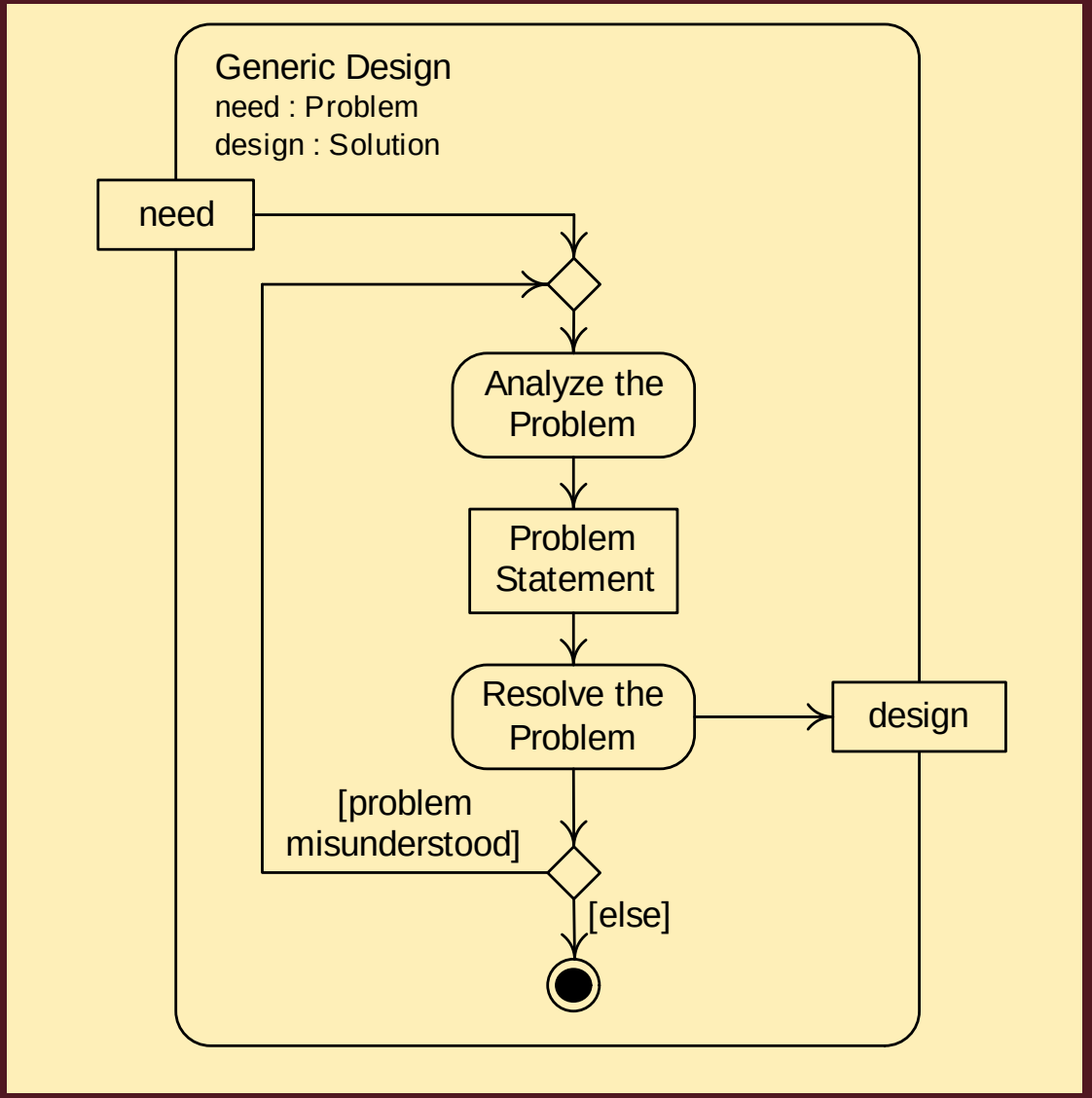


A Problem-Solving Process

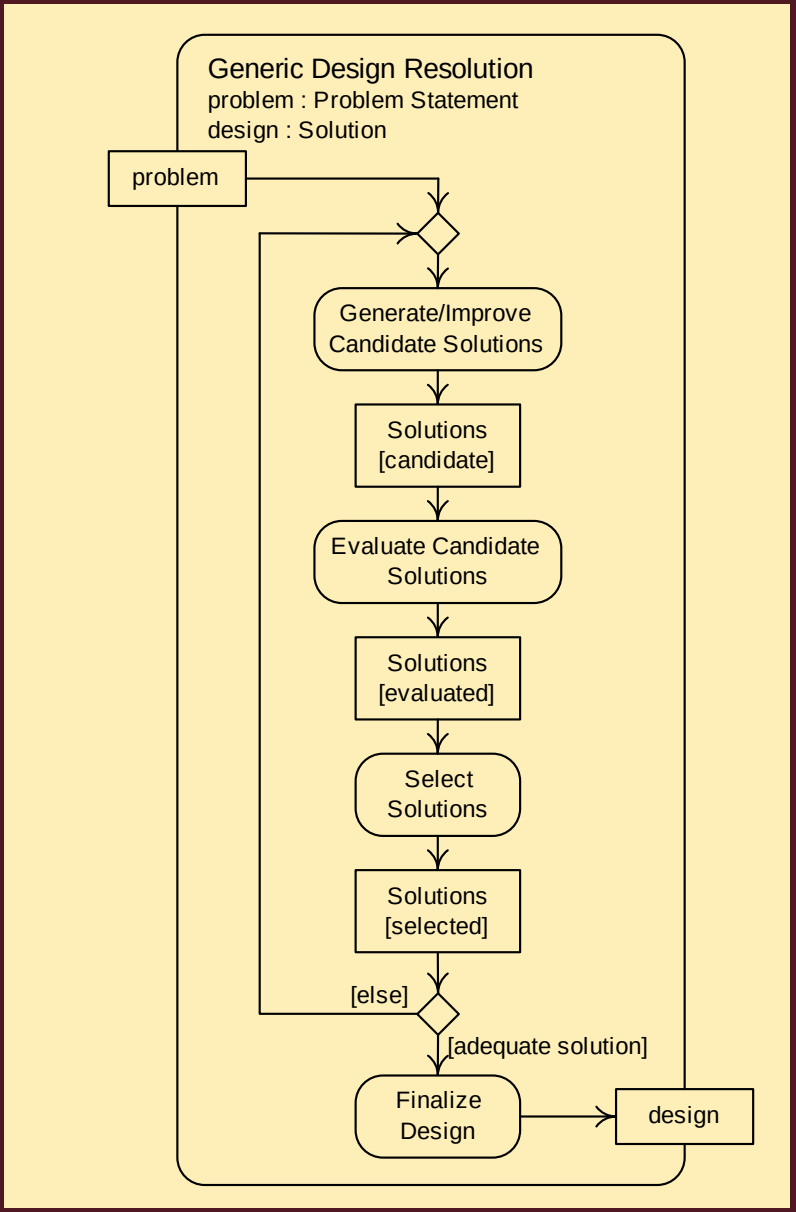
- Understand the problem
- Generate candidate solutions
- Evaluate candidate solutions
- Select the best solution(s)
- Iterate if no solution is adequate
- Ensure the solution is complete and well-documented, and deliver it

- E.g. Specifying class schedule in UBT

A Generic Design Process



A Design Resolution Process



Resolve the Problem

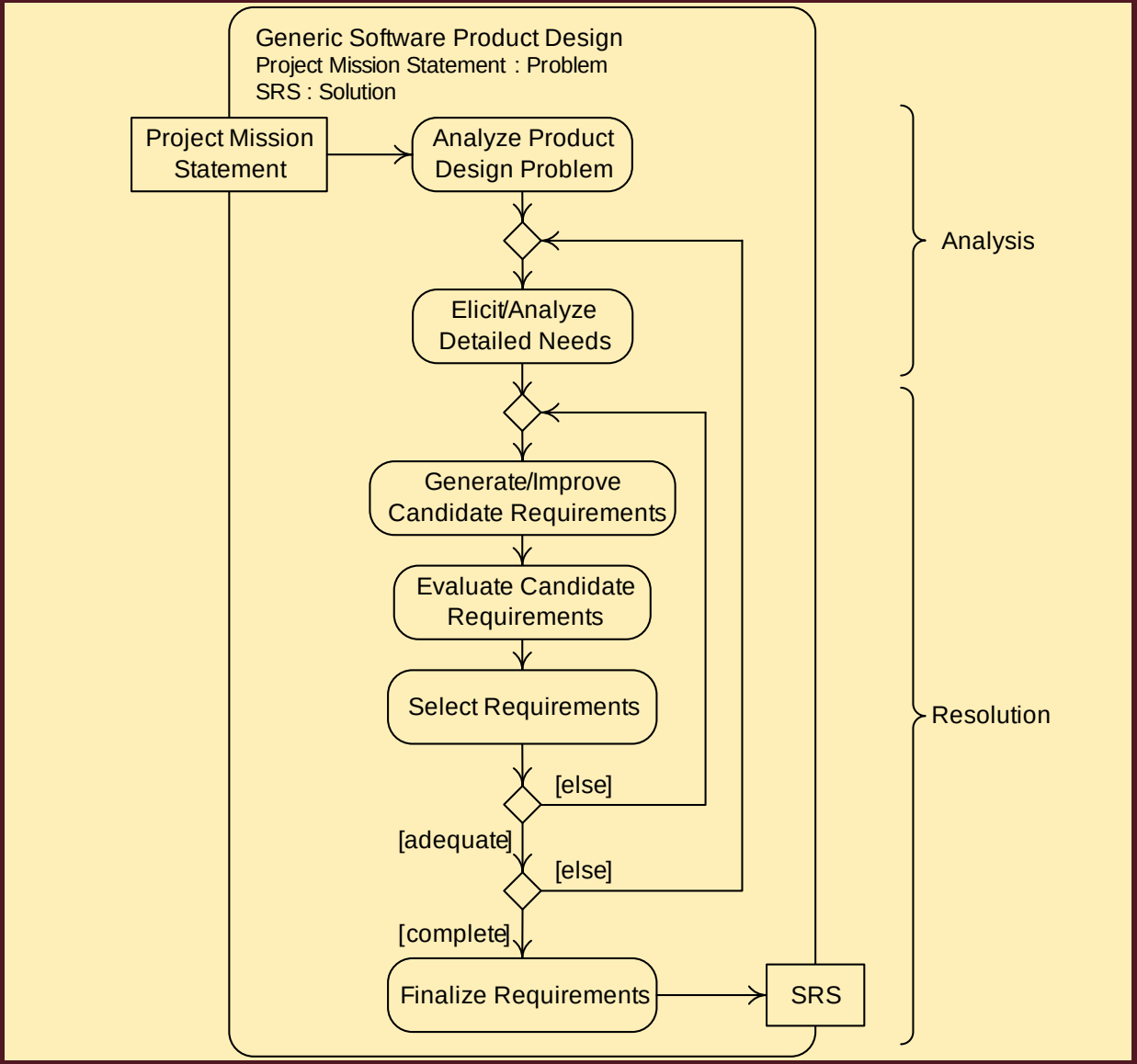
This diagram shows details of the resolution activity from the previous diagram.

Design Process Characteristics

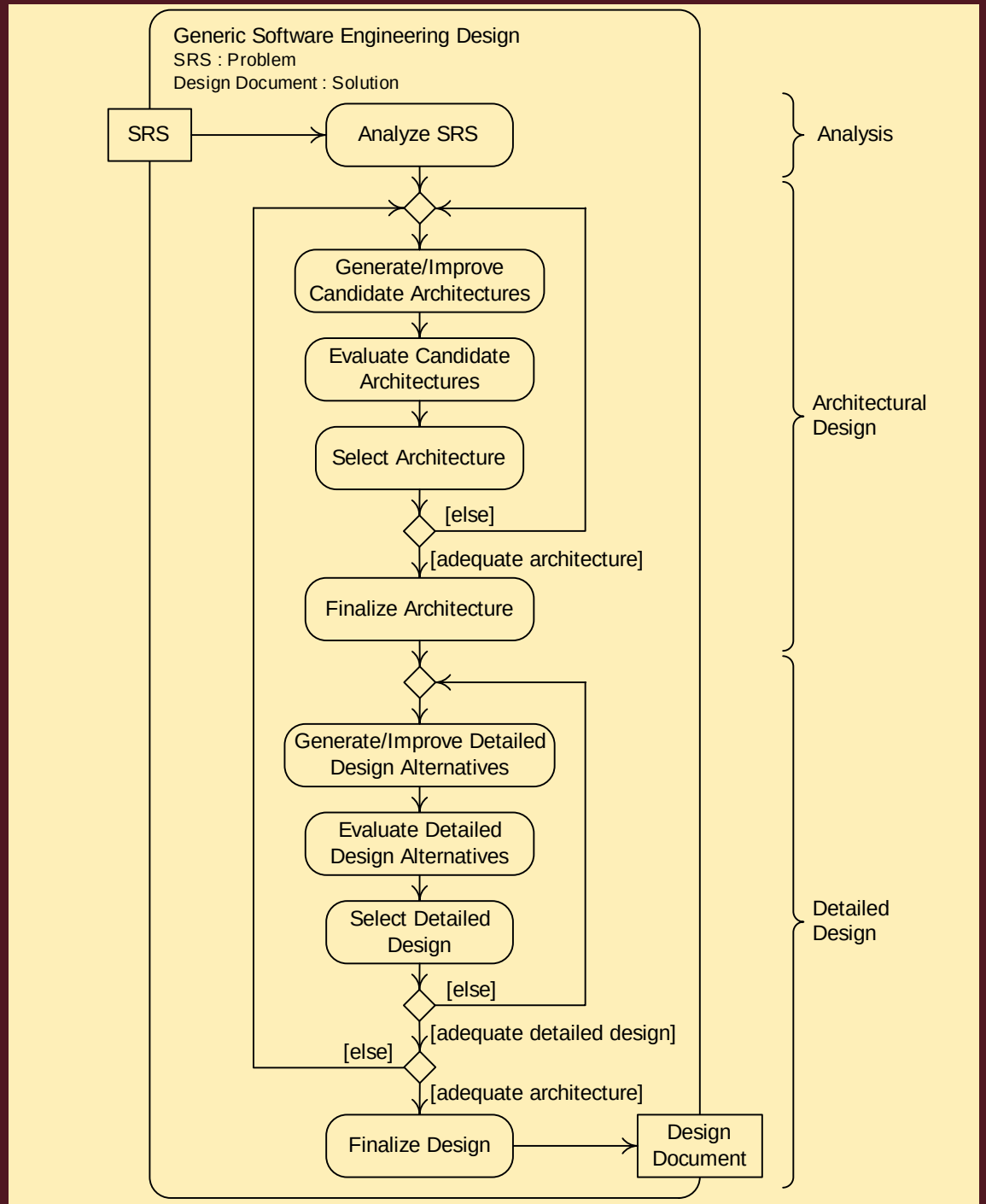
- The best solutions are rarely the first solutions designers think of.
  - *Designers should generate many candidate solutions*.

- The design process is **highly iterative**.
  - *Designers must frequently reanalyze the problem and must generate and improve solutions many times*.

A Generic Software Product Design Process



A Generic Software Engineering Design Process



Architectural and Detailed Design

Architectural design is high-level software engineering design resolution.

Detailed design is low-level software engineering design resolution.

Later discussions will consider architectural and detailed design, the latter being further divided into mid-level and low-level detailed design.

Project Management Activities

1. *Planning* — Formulating a scheme for doing a project.
2. *Organizing* — Structuring the organizational entities involved in a project and assigning them authority and responsibilities.
3. *Staffing* — Filling the positions in an organizational structure and keeping them filled.
4. *Tracking* — Observing the progress of work and adjusting work and plans accordingly.
5. *Leading* — Directing and supporting people doing project work.

Project Planning

- **Estimation** is calculation of the approximate cost, effort, time or resources required to achieve some end.
- A **schedule** specifies the start and duration of work tasks.
- Tasks are allocated resources based on the schedule and estimates.
- **Risk analysis** is an orderly process of identifying, understanding, and assessing **risks** (any occurrence with negative consequences).
- Policies, procedures, tools, and techniques are specified to govern work.

Project Tracking

- Projects may not go as planned for many reasons.
  - Resource consumption is not as expected.
  - Tasks do not take as long as expected.
  - Policies, procedures, tools, or techniques cause problems.
  - Something bad occurs (illness, budget cuts, equipment failures, etc.
- When plans fail they must be adjusted.

Leading a Project

- Direction is needed to follow plans, use resources efficiently, etc.
- Directing people is not enough—people need inspiration, help, a congenial work environment, emotional support, etc.

Iterative Planning and Tracking

- Good **planning** requires knowledge of tasks and their costs, risks, and other details not known until the project is under way—but this is **not known** when plans are made.
- **Iterative planning and tracking** is making a rough base or initial project plan, and refining it at fixed periods during a project in light of tracking data and completed work products.

Design Project Management

- All five project management activities are needed to manage a design project.
- Iterative planning and tracking is the best approach to planning and tracking.
- The design project decomposition on the next slide is useful for planning, organization, staffing, and tracking.
- Design constitutes the largest activity in software development, so **design can drive an entire development project**.

Design Project Decomposition

Work Phase		Typical Work Products
Product Design	Analysis: Design Problem	Statement of interested parties, product concept, project scope, markets, business goals Models (of the problem) Prototypes (exploring the problem)
	Analysis: Detailed Needs	Client surveys, questionnaires, interview transcripts, etc. Problem domain description Lists of needs, stakeholders Models (of the problem) Prototypes (exploring needs)
	Resolution: Product Specification	Requirements specifications Models (of the product) Prototypes (demonstrating the product)
Engineering Design	Analysis	Models (of the engineering problem) Prototypes (exploring the problem)
	Resolution: Architectural Design	Architectural design models Architectural design specifications Architectural prototypes
	Resolution: Detailed Design	Detailed design models Detailed design specifications Detailed design prototypes

Summary

- **Analysis** is breaking a design problem down to understand it; **resolution** is solving a design problem.
- Design processes begin with analysis and have a highly iterative resolution phase.
- Designers should generate many **candidate solutions** and expect to reanalyze and resolve the problem repeatedly.
- Design management is project management and hence requires planning, organization, staffing, tracking, and leadership.
- Iterative planning and tracking is the best way to make and revise plans during a project.

Suggested reading

- Introduction to Software Engineering Design Processes, Principles, and Patterns with UML2
  - **2.2 Software Design Processes**
  - **2.3 Software Design Management**

Thank you!

- Questions